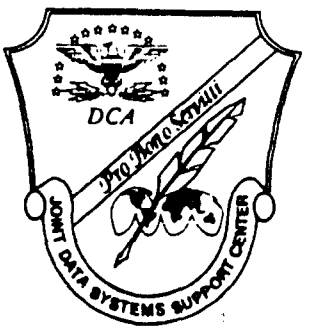
(2)

SOFTWARE REQUIREMENTS
SPECIFICATION
SRS 1-90
VOLUME I
1 DECEMBER 1990



JOINT DATA SYSTEMS
SUPPORT CENTER

SOFTWARE REQUIREMENTS SPECIFICATION FOR THE MAPPING AND GRAPHIC INFORMATION CAPABILITY (MAGIC) VOLUME I - HUMAN INTERFACE CSCI

RECORD OF CHANGES

Change Number	Dated	Date Entered	Signature of Person Making Change

DCA FORM 65
MAR 87

JOINT DATA SYSTEMS SUPPORT CENTER

Software Requirements Specification SRS 1-90

1 December 1990

SOFTWARE REQUIREMENTS SPECIFICATION

FOR THE

MAPPING AND GRAPHIC INFORMATION CAPABILITY (MAGIC)

VOLUME I - HUMAN INTERFACE CSCI

SUBMITTED BY:

Velma R. Blue

for Jeanne L. Muenzen
Acting Chief
Information Systems Branch

APPROVED BY:

Thomas R. Epperson

Thomas R. Epperson
Deputy Director
NMCS ADP Directorate

ACKNOWLEDGMENT

This Software Requirements Specification was prepared under the general direction of the Chief, Information Systems Branch (JNGG); Chief, General Applications Division (JNG); and the Deputy Director, NMCS ADP Directorate (JN).

CONTENTS

TABLES

ABSTRACT

This Software Requirements Specification (SRS) specifies the engineering and qualification requirements for the Human Interface CSCI of the Mapping and Graphic Information Capability (MAGIC). Furthermore, this specification will be used as the basis for the design and formal testing of that CSCI.

The SRS is divided into 3 major sections. These sections cover Engineering Requirements (Section 3), Qualification Requirements (Section 4), and Preparation for Delivery (Section 5).

This specification supersedes both the Rational-generated Software Requirements Specification (configuration identifier 8734/89-SRS-HI-003) and the Interface Requirements Specification (configuration identifier 8734/89-IRS-GIPSY-003) for the Human Interface CSCI that was delivered under Contract Number DCA100-89-C-0015 and dated 13 September 1989.

SECTION 1. SCOPE

This section provides an introduction to the specification. The following paragraphs discuss the identification of the Computer Software Configuration Item (CSCI), provide an overview of the CSCI, and a document overview.

1.1 Identification

This Software Requirements Specification (SRS) establishes the engineering and qualification requirements for the Human Interface CSCI (CSCI-1).

1.2 CSCI Overview

The Human Interface CSCI functions as the logical hub of all MAGIC processing activities and presents a user-friendly graphical user interface (GUI) to the user that is compliant with Open Software Foundation (OSF)/Motif. The user may choose menu selections with either keystrokes or mouse clicks and dialog boxes are provided for those instances when the user is required to provide additional information before MAGIC can continue processing.

When a user initiates a MAGIC session, the program menus are presented to the user, and control of program actions begins in this CSCI. As users navigate through the menu system to an activity they want to perform, control is eventually passed to the appropriate functional CSCI (or Commercial Off-The Shelf (COTS) package), such as Data Management or Business Graphics, to perform the desired activity. The Human Interface CSCI also provides context-sensitive help that is compliant with Motif applications.

1.3 Document Overview

This SRS specifies the requirements allocated to the Human Interface CSCI and enables the Government to assess whether or not the completed CSCI complies with those requirements. Upon Government approval and authentication, the SRS becomes the Allocated Baseline for the CSCI and is used by the contractor as the basis for development and formal testing of the CSCI.

As such, this SRS specifies the complete list of requirements (functional, interface, performance, qualification, etc.) for the Human Interface CSCI. It includes requirements for programming design, adaptation, quality factors, and traceability of the CSCI, as well as delivery preparation and ancillary notes, such as references and terms and abbreviations.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 2. APPLICABLE DOCUMENTS

This section specifies the applicable reference documents that have been used during the preparation of this specification.

2.1 Government Documents

The following documents of the exact issue shown form a part of this specification to the extent specified herein. In the event of conflict between the documents referenced herein and this specification, this specification shall be considered a superseding requirement.

SPECIFICATIONS:

DI-MCCR-80025A	Software Requirements Specifications Data Item Description (DID)
SDP 2-90	Software Development Plan (SDP) for the Mapping and Graphic Information Capability System (MAGIC)
<reference>	Functional Description for the Graphic Information Presentation System (GIPSY)
<reference>	Software Quality Program Plan for the Mapping and Graphic Information Capability System (MAGIC)

STANDARDS:

DOD-STD-2167A	Defense System Software Development

DRAWINGS:

None

OTHER PUBLICATIONS:

PM 1-90	Documentation Standards and Publications Style Manual
TM 405-90	Software Standards and Procedures Manual for the JNGG Graphics Program

Copies of the specifications, standards, drawings, and publications required by suppliers in connection with specified procurement functions should be obtained from the contracting agency or as directed by the contracting officer.

2.2 Non-Government Documents

The following documents of the exact issue shown form a part of this specification to the extent specified herein. In the event of conflict

between the documents referenced herein and the this specification, this specification shall be considered a superseding requirement.

SPECIFICATIONS:

None

STANDARDS:

ANSI X3.159-1989 Programming Language C

DRAWINGS:

None

OTHER PUBLICATIONS:

MIT/LCS/TR-368 The X Window System

<reference> OSF/Motif Users Guide

Technical society and technical association specifications and standards are generally available for reference from libraries. They are also distributed among technical groups and using Federal Agencies.

SECTION 3. ENGINEERING REQUIREMENTS

This section specifies the engineering requirements necessary to ensure proper development of the Human Interface CSCI. All requirements included in this section are allocated from those defined in appendix C of the Functional Description (FD) referenced in the specifications of paragraph 2.1.

3.1 CSCI External Interface Requirements

The Human Interface CSCI interfaces with Data Management, Geographic Mapping, Graphic Editor, Slide Show, and the Internal Processing CSCIs as well as the C Library and Motif. The following subparagraphs provide a general description of each interface.

3.1.1 Human Interface to Data Management (INT-1.001). This interface is used to invoke the capabilities of the Data Management CSCI. This interface enables the user to directly interact with MAGIC's Data Base Management System (DBMS). By using the event-driven X Windows and OSF/Motif, the Human Interface CSCI becomes a controller for the rest of MAGIC.

This interface satisfies the following functional requirements:

a. Use OSF/Motif to provide the graphical user interface (A.11)

b. Identify the user's data file (B.1)

c. Describe data records (B.2)

    (1) FDT (B.2.a)

    (2) Adding to the Index File (B.2.b)

    (3) Augmenting an existing File (B.2.c)

        (a) Extended fields (B.2.c.1)

        (b) Global fields (B.2.c.2)

        (c) Qualify fields (B.2.c.3)

        (d) Specific field references (B.2.c.4)

d. Identify any conditional expressions (B.3)

e. Identify any arithmetic expressions (B.4)

f. Modify data (B.6)

    (1) In-line modification (B.6.a)

(2) User subroutine modification (B.6.b)

(3) Record output table (B.6.c)

(4) Field table (B.6.d)

g. Manipulate data (B.7)

(1) Modify QDT (B.7.a)

(2) Add new fields (B.7.b)

(3) Sort QDF (B.7.c)

(4) Qualify data (B.7.d)

(5) Field table QDF (B.7.e)

(6) Field table qualify (B.7.f)

(7) Field table call (B.7.g)

h. Populate a database or a data file (B.8).

3.1.2 Human Interface to Geographic Mapping (INT-1.002). This interface is used to invoke the capabilities of the Geographic Mapping CSCI. This interface enables the user to directly interact with MAGIC's mapping capabilities provided by function calls to the DeLorme Mapping System (DMS).

This interface satisfies the following functional requirement--use OSF/Motif to provide the graphical user interface (A.11). By using the event-driven X Windows and OSF/Motif, the Human Interface CSCI becomes a controller for the rest of MAGIC.

3.1.3 Human Interface to Graphic Editor (INT-1.003). This interface is used to invoke the capabilities of the Graphic Editor CSCI. This interface enables the user to interactively modify graphical screen displays.

This interface satisfies the following functional requirement--use OSF/Motif to provide the graphical user interface (A.11). By using the event-driven X Windows and OSF/Motif, the Human Interface CSCI becomes a controller for the rest of MAGIC.

3.1.4 Human Interface to Slide Show (INT-1.004). This interface is used to invoke the capabilities of the Slide Show CSCI. This interface enables the user to develop and execute briefing presentations on the workstation.

This interface satisfies the following functional requirement--use OSF/Motif to provide the graphical user interface (A.11). By using the event-driven X Windows and OSF/Motif, the Human Interface CSCI becomes a controller for the

rest of MAGIC.

3.1.5 Human Interface to Internal Processing (INT-1.005). This interface is used by the Human Interface CSCI to access low-level and system-wide utilities and services resident in the Internal Processing CSCI (e.g., path name manipulation, string manipulation, Unix system toolbox routines, and modem toolbox control).

This interface satisfies the following functional requirements:

a. Use OSF/Motif to provide the graphical user interface (A.11)

b. Perform file management (G.1)

c. Control input/output operations (G.2)

d. Save and recall all information necessary to start a new GIPSY session from the departure point of the current GIPSY session (G.3)

e. Convert the qualified data and its internal matrix version to Data Interchange Format (DIF) (G.4)

f. Control various devices such as terminals, printers, or plotters via device drivers (G.7)

g. Request operating system services (G.8)

h. Perform specialized processing by executing system-supplied and user subroutines (G.9)

i. Allowing certain globals to prevail throughout a user session (G.10)

   (1) Command line options (G.10.a)

   (2) Module transfer (G.10.b)

   (3) Classification markings (G.10.c)

   (4) Report titles and modifying them (G.10.d)

   (5) Clearing specific commands (G.10.e)

   (6) Size of text (G.10.f)

   (7) Color of text (G.10.g)

j. Color processing (G.11)

k. User control of operating environment attributes (G.12)

l. Identifying GIPSY's collective input statements (G.13)

(1) PCS (G.13.a)

(2) Clear PCS (G.13.b)

(3) Save PCS (G.13.c)

(4) RETURN statement (G.13.d)

m. Identify and save GIPSY's internal data structures (G.14)

(1) FDT (G.14.a)

(2) QDF (G.14.b)

(3) QDT (G.14.c)

(4) DAFC (G.14.d)

(5) GDS (G.14.e)

n. Executing user-supplied subroutines (G.15)

o. Executing other TSS or JDAC commands (G.16).

3.1.6 Human Interface to C Library (INT-1.006). This interface is used to access standard C language functions and to utilize C types.

This interface satisfies the following functional requirements:

a. Initialize MAGIC (A.1)

b. Parse user selections, choices, and language input (A.2)

c. Recognize interactive operating system commands (i.e., JDAC & TSS commands) (A.3)

d. Define and name a group of GIPSY statements which may be subsequently executed through reference to the defined name via an application (i.e., DO PROCESS) (A.4)

e. Execute a GIPSY application which prompts the user for input (A.5)

f. Perform interactive error detection and handling (A.6)

g. Provide an on-line interactive help facility (A.7)

h. Allocate and initialize the DAFC (A.8)

i. Establish workstation environment to X Windows (A.9)

j. Access statistics file (A.10)

k. Use OSF/Motif to provide the graphical user interface (A.11).

3.1.7 Human Interface to Motif (INT-1.007). This interface is used to access the capabilities of Motif to provide the GUI and access to underlying X Windows functionality.

This interface satisfies the following functional requirements:

a. Initialize MAGIC (A.1)

b. Parse user selections, choices, and language input (A.2)

c. Perform interactive error detection and handling (A.6)

d. Provide an on-line interactive help facility (A.7)

e. Establish workstation environment to X Windows (A.9)

f. Use OSF/Motif to provide the graphical user interface (A.11).

3.2 CSCI Capability Requirements

The following subparagraphs identify the capability requirements that the Human Interface CSCI shall satisfy. The CSCI operates only in the assisted state meaning that a fully-functional GUI is being used. The unassisted state is not applicable to this CSCI by virtue of the functionality provided. The assisted state has two modes--local and remote. A correlation of the CSCI's capabilities to both states and their modes is depicted in table 3-1.

3.2.1 GUI Processing (CAP-1.1). This capability provides the facility for MAGIC to interact with the user by means of a Motif-based GUI. Major functionality supported by this capability includes: (1) system initialization, (2) the creation of menus, screens, and dialog boxes that permit the user to make selections with either keystroke or mouse click, (3) the processing of user input to include validation and verification of that input, and (4) calling other MAGIC CSCIs and COTS packages as needed to satisfy user requests for action. The scope of this capability shall also include the recognition that a user wishes to: (1) bypass normal GUI processing and work directly with a COTS package or (2) use xterm to communicate with the host platform.

This capability satisfies the following functional requirements:

a. Initialize MAGIC (A.1)

b. Parse user selections, choices, and language input (A.2)

Table 3-1. Mapping of States/Modes to Capabilities

STATE	MODE	CAP-1.1	CAP-1.2	CAP-1.3	CAP-1.4
ASSISTED	LOCAL	●	●	●	●
	REMOTE	●	●	●	●
UNASSISTED	LOCAL				
	REMOTE				

c. Recognize interactive operating system commands (i.e., JDAC & TSS commands) (A.3)

d. Define and name a group of GIPSY statements which may be subsequently executed through reference to the defined name via an application (i.e., DO PROCESS) (A.4)

e. Execute a GIPSY application which prompts the user for input (A.5)

f. Perform interactive error detection and handling (A.6)

g. Provide an on-line interactive help facility (A.7)

h. Allocate and initialize the DAFC (A.8)

i. Establish workstation environment to X Windows (A.9)

j. Access statistics file (A.10)

k. Use OSF/Motif to provide the graphical user interface (A.11).

3.2.2 Help Facility (CAP-1.2). This capability provides the means for MAGIC to provide a user-friendly, context-sensitive help facility to the user. Help is requested through either keystroke or mouse click from the standard Motif-based GUI screens unless superseded by a COTS package or xterm processing.

This capability satisfies the following functional requirements:

a. Parse user selections, choices, and language input (A.2)

b. Provide an on-line interactive help facility (A.7)

c. Use OSF/Motif to provide the graphical user interface (A.11).

3.2.3 Error Handling (CAP-1.3). This capability provides the means for MAGIC to perform interactive error handling and correction when detected through normal processing. Error handling is provided with the standard Motif-based GUI screens unless superseded by a COTS package or xterm processing.

This capability satisfies the following functional requirements:

a. Parse user selections, choices, and language input (A.2)

b. Perform interactive error detection and handling (A.6)

c. Use OSF/Motif to provide the graphical user interface (A.11).

3.2.4 Motif Tools (CAP-1.4). This capability provides the Motif-based tools that shall be developed to assist in the creation and utilization of Motif-compliant screens. Motif is comprised of routines that group lower-level X

Windows routines (from Xlib) into a higher level of abstraction. This capability will take the process one step further and contain tools built from Motif routines that are used extensively throughout the Human Interface CSCI.

This capability satisfies requirement A.11--Use OSF/Motif to provide the graphical user interface.

3.3 CSCI Internal Interfaces

The following subparagraphs provide a description of the internal interfaces of the Human Interface CSCI.

3.3.1 GUI Processing to Help Facility (RIN-1.001). This interface is used by GUI Processing (CAP-1.1) to request the on-line help functionality to satisfy a user's request. Help requests issued by the user while utilizing a COTS package or within xterm (host-based access) would be handled by local help functions wherever appropriate.

This internal interface satisfies the following functional requirements:

a. Initialize MAGIC (A.1)

b. Parse user selections, choices, and language input (A.2)

c. Recognize interactive operating system commands (i.e., JDAC & TSS commands) (A.3)

d. Define and name a group of GIPSY statements which may be subsequently executed through reference to the defined name via an application (i.e., DO PROCESS) (A.4)

e. Execute a GIPSY application which prompts the user for input (A.5)

f. Perform interactive error detection and handling (A.6)

g. Provide an on-line interactive help facility (A.7)

h. Allocate and initialize the DAFC (A.8)

i. Establish workstation environment to X Windows (A.9)

j. Access statistics file (A.10)

k. Use OSF/Motif to provide the graphical user interface (A.11).

3.3.2 GUI Processing to Error Handling (RIN-1.002). This interface is used by GUI Processing (CAP-1.1) to request error processing functions as a result of normal user error. Errors detected while using a COTS package or within xterm (host-based access) would be handled by local error processing functions.

This internal interface satisfies the following functional requirements:

a. Initialize MAGIC (A.1)

b. Parse user selections, choices, and language input (A.2)

c. Recognize interactive operating system commands (i.e., JDAC & TSS commands) (A.3)

d. Define and name a group of GIPSY statements which may be subsequently executed through reference to the defined name via an application (i.e., DO PROCESS) (A.4)

e. Execute a GIPSY application which prompts the user for input (A.5)

f. Perform interactive error detection and handling (A.6)

g. Provide an on-line interactive help facility (A.7)

h. Allocate and initialize the DAFC (A.8)

i. Establish workstation environment to X Windows (A.9)

j. Access statistics file (A.10)

k. Use OSF/Motif to provide the graphical user interface (A.11).

3.3.3 GUI Processing to Motif Tools (RIN-1.003). This interface is used by GUI Processing (CAP-1.1) to access the Motif-based tools built for use throughout the Human Interface CSCI.

This internal interface satisfies the following functional requirements:

a. Initialize MAGIC (A.1)

b. Parse user selections, choices, and language input (A.2)

c. Recognize interactive operating system commands (i.e., JDAC & TSS commands) (A.3)

d. Define and name a group of GIPSY statements which may be subsequently executed through reference to the defined name via an application (i.e., DO PROCESS) (A.4)

e. Execute a GIPSY application which prompts the user for input (A.5)

f. Perform interactive error detection and handling (A.6)

g. Provide an on-line interactive help facility (A.7)

h. Allocate and initialize the DAFC (A.8)

i. Establish workstation environment to X Windows (A.9)

j. Access statistics file (A.10)

k. Use OSF/Motif to provide the graphical user interface (A.11).

3.3.4 Help Facility to Motif Tools (RIN-1.004). This interface is used by Help Facility (CAP-1.2) to access the Motif-based tools built for use throughout the Human Interface CSCI.

This internal interface satisfies the following functional requirements:

a. Parse user selections, choices, and language input (A.2)

b. Provide an on-line interactive help facility (A.7)

c. Use OSF/Motif to provide the graphical user interface (A.11).

3.3.5 Error Handling to Motif Tools (RIN-1.005). This interface is used by Error Handling (CAP-1.3) to access the Motif-based tools built for use throughout the Human Interface CSCI.

This internal interface satisfies the following functional requirements:

a. Parse user selections, choices, and language input (A.2)

b. Perform interactive error detection and handling (A.6)

c. Use OSF/Motif to provide the graphical user interface (A.11).

3.4 CSCI Data Element Requirements

No internal or external data elements have been identified for this CSCI.

3.5 Adaptation Requirements

The following subparagraphs specify the requirements for adapting this CSCI to site-unique conditions and to changes in the system environment.

3.5.1 Installation-Dependent Data. There are no specific installation-dependent data requirements needed for adapting this CSCI to site-unique conditions or to changes in the system environment.

3.5.2 Operational Parameters. There are no specific operational parameters needed for adapting this CSCI to site-unique conditions or to changes in the system environment.

3.6 Sizing and Timing Requirements

Sizing requirements pertinent to this CSCI are:

a. A minimum of 8 megabytes (Mb) of Random Access Memory (RAM) shall be required to execute MAGIC.

b. A minimum of 2 Mb of free disk space shall be required to execute MAGIC.

c. A minimum of 16 Mb of swap space shall be required to execute MAGIC.

Timing requirements pertinent to this CSCI are two-fold:

a. MAGIC's response to a user's mouse click or a keystroke for a menu or dialog box shall be within a 5 second timeframe.

b. If the user-input choice requires MAGIC to interface with a COTS package (either launching or processing), system response shall be within a 5 second timeframe. In other words, the user must either receive some sort of acknowledgment that processing is going on or obtain the end result of their selection.

3.7 Safety Requirements

This CSCI is a software product and is intended for use in an office environment. As such, there are no applicable requirements regarding potential hazards to personnel, property, and the physical environment.

3.8 Security Requirements

MAGIC is released as an unclassified system and all system files released with it are unclassified. However, MAGIC's features may be used to analyze and present classified information from classified databases. Under these circumstances, MAGIC shall provide the facilities to properly label the screen images and the hardcopy reports, but it is and will remain the user's responsibility to safeguard any and all classified information. MAGIC cannot grant access to classified databases unless the user has permission and access to those files.

Security requirements for all hardware suites and configurations capable of executing MAGIC shall remain the same as required for other operational considerations pertinent and applicable to that equipment and environment. Furthermore, the safeguarding of privacy act information also remains the user's responsibility.

Additional requirements regarding integrity requirements are specified in subparagraph 3.10.4 of this specification.

3.9 Design Constraints

This CSCI will be developed in accordance with the standards identified in the Software Standards and Procedures Manual (SSPM). MAGIC has very few design constraints due to its utilization of ANSI C, X Windows, and OSF/Motif in functional processing:

a. Due to usage of the Oracle COTS package for database management processing, MAGIC is constrained to those data types and parameters supported by Oracle's SQL*Loader package.

b. Specific tables stored in the Oracle database (on the workstation) as well as specific data files resident on the WWMCCS host are accessible only to the MAGIC user who has created them (or to one who has been given permissions to them by the owner).

c. Usage of the host-based GIPSY system will introduce a number of limitations that do not apply to a MAGIC user utilizing workstation-based data. Specifically, not all of Oracle's capabilities supported by MAGIC in local mode can be supported by MAGIC's interface to GIPSY due to inherent differences between the two systems (Oracle and GIPSY). The user must be at least somewhat aware of GIPSY concepts and terminology which is different (e.g., File Descriptor Table (FDT) and Index File) and not all functionality can be supported (e.g., very limited Oracle GROUP functionality).

d. Usage of a modem for host access will have definitive impacts related to both how and how fast MAGIC can access the host, retrieve the data, and make it available to the MAGIC user on the workstation. Some software developed for the modem will be modem-specific and some will be inapplicable when MAGIC is transitioned to a direct host communications connection. The processing speed by which MAGIC users can receive response from the host and obtain their data is directly linked to modem speed (currently 2400 baud) and access availability (via Defender).

e. Target workstation hardware and operating system specifics are still changing at the time of writing this SRS. Since a prototype is being developed on a Sun Scalable Processor Architecture (SPARC) station and the target is presumed to be the Macintosh IIfx, the design is limited to those aspects common across the platforms wherever possible.

f. The utilization of the Wingz COTS package to perform nearly all business graphics-related processing introduces several design constraints. Currently, nearly all of the constraints noted below arise from the fact that MAGIC is being developed on the Sun SPARCstation and the Wingz version (Version 1.0) for the Sun platform was designed for execution in the SunView environment. Since MAGIC has been designed for the X Windows environment, a method was found

that permits the execution of Wingz under the X11/NeWS server with the following design constraints:

(1) The "look and feel" of Wingz is not consistent with MAGIC's Motif-based "look and feel."

(2) The help text available with Wingz (in Version 1.0) cannot be modified.

(3) The menu bar title cannot be modified.

(4) The proper import of data into Wingz can be guaranteed only by using an assisted query.

(5) Curve graphs, Gantt charts, and histograms are not directly supported by Wingz

(6) Wingz requires a PostScript-capable printer or Hewlett-Packard Graphic Language (HPGL) plotter to print.

(7) The experimental interface to the X11/NeWS server may cause unpredictable results.

(8) The code generated to support both the X11/NeWS server execution method may not be portable to other environments.

3.10 Software Quality Factors

The following subparagraphs specify the software quality factors or "fitness for use" characteristics that are required for the Human Interface CSCI. They are divided into 11 categories: correctness, reliability, efficiency, integrity, usability, maintainability, testability, flexibility, portability, reusability, and interoperability.

3.10.1 Correctness Requirements. The requirements contained in this subparagraph specify the extent to which the CSCI is expected to satisfy its specifications and fulfill the user's mission objectives. The correctness requirements are:

a. The software shall be traceable. The functionality of the CSCI must possess a clear linkage from the requirements to the implementation with respect to the specific development and operational environment.

b. The software shall be consistent. The contractor is required to provide uniform design and implementation of techniques and notation.

c. The software shall be complete. The functionality of the CSCI must provide a full implementation of the functions required.

3.10.2 Reliability Requirements. The requirements contained in this subparagraph specify the extent to which the CSCI is expected to perform its intended functions with required precision. The reliability requirements are:

a. The error tolerance of the software shall be 2 percent. The CSCI is required to provide continuity of operation at least 98 percent of the time.

b. The software shall be consistent. The contractor is required to provide uniform design and implementation of techniques and notation.

c. The software shall be accurate. The software must provide the user's required precision in calculations and outputs within the limitations of the various COTS packages being used.

d. The software shall be simplistic. The functions of the CSCI must be implemented in a most understandable manner and avoid those coding/implementation practices that increase complexity.

3.10.3 Efficiency Requirements. The requirements contained in this subparagraph specify the amount of computing resources and code required by the CSCI to perform its functions. The efficiency requirements are:

a. The execution efficiency of the software shall be in accordance with the timing requirements of paragraph 3.6.

b. The storage efficiency of the software shall be in accordance with the sizing requirements of paragraph 3.6.

3.10.4 Integrity Requirements. The requirements contained in this subparagraph specify the extent to which access to the CSCI's software or data by unauthorized persons should be controlled. The integrity requirements are:

a. The CSCI shall be access controlled. To the extent provided by the workstation environment, only authorized users shall be permitted to access and execute this CSCI.

b. The software shall be access auditable. Some methodology must be provided for an audit of the access of both software and data.

3.10.5 Usability Requirements. The requirements contained in this subparagraph specify the effort required to learn, operate, prepare, input, and interpret the output of this CSCI. The usability requirements are:

a. Training for the use of this CSCI shall be provided as required through normal User Support activities which include functional demonstrations. Formal training is not required at this time due to the requirements for user-friendliness and usability satisfied by this CSCI.

b. The software shall be communicative and provide useful inputs and outputs which can be assimilated by the user.

c. The software shall be operable. A smooth transition from current GIPSY operations as well as initial familiarizations with the Unix-based workstation must be provided wherever appropriate.

3.10.6 Maintainability Requirements. The requirements contained in this subparagraph specify the effort required to locate and fix an error in the operational software. The maintainability requirements are:

a. The software shall be consistent. The contractor is required to provide uniform design and implementation of techniques and notation.

b. The software shall be simplistic. The functions of the CSCI must be implemented in a most understandable manner and avoid those coding/implementation practices that increase complexity.

c. The software shall be concise. Functions must be implemented with a minimum amount of code.

d. The software shall be modular. The modularity of the CSCI shall be designed and implemented using four major attributes:

(a) Cohesiveness refers to the functional strength of a module, or how single-minded a module is. The modules shall strive for high cohesion (functional) wherever possible although mid-range cohesion is acceptable. The seven types of module cohesion are:

(1) Coincidental cohesion (WORST)

(2) Logical cohesion

(3) Temporal cohesion

(4) Procedural cohesion

(5) Communicational cohesion

(6) Informational cohesion

(7) Functional cohesion (BEST).

(b) Coupling refers to the interdependence of modules (i.e., how they communicate with each other). Of the six types of coupling, modules shall strive to employ data coupling wherever possible. The types of module coupling are:

(1) Content coupling (WORST)

(2) Common Coupling

(3) External Coupling

(4) Control Coupling

(5) Stamp Coupling

(6) Data Coupling (BEST).

(c) Complexity refers to the logical or control flow complexity of any given module. Modules shall be designed with low complexity since they will be easier to test and maintain:

(1) The cyclomatic complexity of a module shall be kept within 10 as determined by McCabe's Cyclomatic Complexity Metric.

(2) The size of any module shall be no more than 200 lines of executable code.

(d) Structure refers to whether or not a program is structured. Modules shall be designed in a structured manner to enhance maintainability as determined by the principles of essential complexity and program "knots":

(1) The essential complexity of a module shall be 1.

(2) Modules shall have 0 "knots." Knots are those places in a program where the control path crosses another.

e. The software shall be self-descriptive. The software must contain sufficient comments to provide explanation of the implementation of a function.

f. The software shall be traceable. The functionality of the CSCI must possess a clear linkage from the requirements to the implementation with respect to the specific development and operational environment.

3.10.7 Testability Requirements. The requirements contained in this subparagraph specify the effort required to test the CSCI to ensure that it performs its intended function. The testability requirements are:

a. The software shall be simplistic. The functions of the CSCI must be implemented in a most understandable manner and avoid those coding/implementation practices that increase complexity.

b. The software shall be modular. The CSCI must satisfy the requirements of modularity specified in subparagraph 3.10.6 above.

c. The software shall support instrumentation. All paths must be

testable and all input parameters must be boundary testable (as defined in the SQPP).

d. The software shall be self-descriptive. The software must contain sufficient comments to provide explanation of the implementation of a function.

3.10.8 Flexibility Requirements. The requirements contained in this subparagraph specify the effort required to modify operational software. The flexibility requirements are:

a. The software shall be modular. The CSCI must satisfy the requirements of modularity specified in subparagraph 3.10.6 above.

b. The software shall be general. The software should not have input, processing, and output functions mixed in the same modules; all constants should be defined only once; and application and machine-dependent functions should not be mixed in the same modules.

c. The software shall be expandable. The CSCI must perform logical processing independent of data storage specifications (not commit all available memory capacity) and be extensible in terms of computational functions.

d. The software shall be self-descriptive. The software must contain sufficient comments to provide explanation of the implementation of a function.

3.10.9 Portability Requirements. The requirements contained in this subparagraph specify the effort required to transfer the CSCI from one hardware configuration and/or software system environment to another. The portability requirements are:

a. The software shall be modular. The CSCI must satisfy the requirements of modularity specified in subparagraph 3.10.6 above.

b. The software shall be self-descriptive. The software must contain sufficient comments to provide explanation of the implementation of a function.

c. The software shall be machine-independent. The ANSI C code used should be independent of word and character size and the data representation should also be machine-independent. Wherever possible, modules should be free of input/output references.

d. The software shall be as software system-independent as possible. The CSCI shall utilize only a common, standard subset of ANSI C and should limit dependence on software system utilities and software system library routines wherever possible. If at all possible, there should be no operating system references.

3.10.10 Reusability Requirements. The requirements contained in this subparagraph specify the extent to which the programs of the CSCI can be used in other applications (related to the packaging and scope of the functions that the programs perform). The reusability requirements are:

a. The software shall be general. The software should not have input, processing, and output functions mixed in the same modules; all constants should be defined only once; and application and machine-dependent functions should not be mixed in the same modules.

b. The software shall be modular. The CSCI must satisfy the requirements of modularity specified in subparagraph 3.10.6 above.

c. The software shall be as software system-independent as possible. The CSCI shall utilize only a common, standard subset of ANSI C and should limit dependence on software system utilities and software system library routines wherever possible. If at all possible, there should be no operating system references.

d. The software shall be machine-independent. The ANSI C code used should be independent of word and character size and the data representation should also be machine-independent. Wherever possible, modules should be free of input/output references.

e. The software shall be self-descriptive. The software must contain sufficient comments to provide explanation of the implementation of a function.

3.10.11 Interoperability Requirements. The requirements contained in this subparagraph specify the effort required to couple this MAGIC CSCI with another system. The interoperability requirements are:

a. The software shall be modular. The CSCI must satisfy the requirements of modularity specified in subparagraph 3.10.6 above.

b. The software shall utilize communications commonality wherever appropriate. It is recognized that this requirement will be satisfied primarily by the Internal Processing CSCI (refer to Volume VII of this SRS).

c. The software shall utilize data commonality. The CSCI should use a single module to perform any data translations and standard data representations should be used.

3.11 Human Performance/Human Engineering Requirements

Issues related to human performance and human engineering concerns have been noted and discussed previously in subparagraph 3.10.5 of this specification.

Operational issues are concerned with the hardware and software support

environments required for the user. A brief summation of the user's operational needs would include the following:

a. Access to a Unix-based color graphics workstation that has both Motif (Release 1.0.a) and X Windows (Release 11, Version 3) installed on it.

b. Access to auxiliary devices such as dot matrix printers, Postscript-capable laser printers, floppy disk drives (1.44 Mb), external tape backup units, and external mass storage devices.

c. Access to the WWMCCS host via xterm on the workstation.

Human error is a final issue related to human engineering requirements. Once Human Interface has been initiated, errors will be captured and handled by this CSCI's error handling facilities.

3.12 Requirements Traceability

A mapping of the engineering requirements in this specification to the functional requirements applicable to this CSCI in the FD is provided in table 3-2. A mapping of the allocation of the CSCI requirements from the FD to the engineering requirements in this specification is provided as table 3-3.

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 1 of 10)

FUNCTIONAL REQUIREMENTS / ENGINEERING REQUIREMENTS	A.1	A.2	A.3	A.4	A.5	A.6	A.7	A.8	A.9	A.10	A.11	B.1	B.2
INT-1.001											●	●	●
INT-1.002											●		
INT-1.003											●		
INT-1.004											●		
INT-1.005											●		
INT-1.006	●	●	●	●	●	●	●	●	●	●	●		
INT-1.007	●	●				●	●		●		●		
CAP-1.1	●	●	●	●	●	●	●	●	●	●	●		
CAP-1.2		●					●				●		
CAP-1.3		●				●					●		
CAP-1.4											●		
RIN-1.001	●	●	●	●	●	●	●	●	●	●	●		
RIN-1.002	●	●	●	●	●	●	●	●	●	●	●		
RIN-1.003	●	●	●	●	●	●	●	●	●	●	●		
RIN-1.004		●					●				●		
RIN-1.005		●				●					●		
SIZING	●	●	●	●	●	●	●	●	●	●	●	●	●
TIMING	●	●	●	●	●	●	●	●	●	●	●	●	●
DESIGN CONSTRAINTS	●	●	●	●	●	●	●	●	●	●	●	●	●
CORRECTNESS	●	●	●	●	●	●	●	●	●	●	●	●	●
RELIABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
EFFICIENCY	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEGRITY	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 2 of 10)

ENGINEERING REQUIREMENTS \ FUNCTIONAL REQUIREMENTS	B.2.a	B.2.b	B.2.c	B.2.c.1	B.2.c.2	B.2.c.3	B.2.c.4	B.3	B.4	B.6	B.6.a	B.6.b	B.6.c
INT-1.001	●	●	●	●	●	●	●	●	●	●	●	●	●
INT-1.002													
INT-1.003													
INT-1.004													
INT-1.005													
INT-1.006													
INT-1.007													
CAP-1.1													
CAP-1.2													
CAP-1.3													
CAP-1.4													
RIN-1.001													
RIN-1.002													
RIN-1.003													
RIN-1.004													
RIN-1.005													
SIZING	●	●	●	●	●	●	●	●	●	●	●	●	●
TIMING	●	●	●	●	●	●	●	●	●	●	●	●	●
DESIGN CONSTRAINTS	●	●	●	●	●	●	●	●	●	●	●	●	●
CORRECTNESS	●	●	●	●	●	●	●	●	●	●	●	●	●
RELIABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
EFFICIENCY	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEGRITY	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 3 of 10)

ENGINEERING REQUIREMENTS \ FUNCTIONAL REQUIREMENTS	B.6.d	B.7	B.7.a	B.7.b	B.7.c	B.7.d	B.7.e	B.7.f	B.7.g	B.8	G.1	G.2	G.3
INT-1.001	●	●	●	●	●	●	●	●	●	●			
INT-1.002													
INT-1.003													
INT-1.004													
INT-1.005											●	●	●
INT-1.006													
INT-1.007													
CAP-1.1													
CAP-1.2													
CAP-1.3													
CAP-1.4													
RIN-1.001													
RIN-1.002													
RIN-1.003													
RIN-1.004													
RIN-1.005													
SIZING	●	●	●	●	●	●	●	●	●	●	●	●	●
TIMING	●	●	●	●	●	●	●	●	●	●	●	●	●
DESIGN CONSTRAINTS	●	●	●	●	●	●	●	●	●	●	●	●	●
CORRECTNESS	●	●	●	●	●	●	●	●	●	●	●	●	●
RELIABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
EFFICIENCY	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEGRITY	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 4 of 10)

FUNCTIONAL REQUIREMENTS / ENGINEERING REQUIREMENTS	G.4	G.7	G.8	G.9	G.10	G.10.a	G.10.b	G.10.c	G.10.d	G.10.e	G.10.f	G.10.g	G.11
INT-1.001													
INT-1.002													
INT-1.003													
INT-1.004													
INT-1.005	●	●	●	●	●	●	●	●	●	●	●	●	●
INT-1.006													
INT-1.007													
CAP-1.1													
CAP-1.2													
CAP-1.3													
CAP-1.4													
RIN-1.001													
RIN-1.002													
RIN-1.003													
RIN-1.004													
RIN-1.005													
SIZING	●	●	●	●	●	●	●	●	●	●	●	●	●
TIMING	●	●	●	●	●	●	●	●	●	●	●	●	●
DESIGN CONSTRAINTS	●	●	●	●	●	●	●	●	●	●	●	●	●
CORRECTNESS	●	●	●	●	●	●	●	●	●	●	●	●	●
RELIABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
EFFICIENCY	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEGRITY	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 5 of 10)

FUNCTIONAL REQUIREMENTS / ENGINEERING REQUIREMENTS	G.12	G.13	G.13.a	G.13.b	G.13.c	G.13.d	G.14	G.14.a	G.14.b	G.14.c	G.14.d	G.14.e	G.15	G.16
INT-1.001														
INT-1.002														
INT-1.003														
INT-1.004														
INT-1.005	●	●	●	●	●	●	●	●	●	●	●	●	●	●
INT-1.006														
INT-1.007														
CAP-1.1														
CAP-1.2														
CAP-1.3														
CAP-1.4														
RIN-1.001														
RIN-1.002														
RIN-1.003														
RIN-1.004														
RIN-1.005														
SIZING	●	●	●	●	●	●	●	●	●	●	●	●	●	●
TIMING	●	●	●	●	●	●	●	●	●	●	●	●	●	●
DESIGN CONSTRAINTS	●	●	●	●	●	●	●	●	●	●	●	●	●	●
CORRECTNESS	●	●	●	●	●	●	●	●	●	●	●	●	●	●
RELIABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●	●
EFFICIENCY	●	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEGRITY	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 6 of 10)

FUNCTIONAL REQUIREMENTS / ENGINEERING REQUIREMENTS	A.1	A.2	A.3	A.4	A.5	A.6	A.7	A.8	A.9	A.10	A.11	B.1	B.2
USABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
MAINTAINABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
TESTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
FLEXIBILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
PORTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
REUSABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEROPER-ABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 7 of 10)

ENGINEERING REQUIREMENTS \ FUNCTIONAL REQUIREMENTS	B.2.a	B.2.b	B.2.c	B.2.c.1	B.2.c.2	B.2.c.3	B.2.c.4	B.3	B.4	B.5	B.6.a	B.6.b	B.6.c
USABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
MAINTAINABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
TESTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
FLEXIBILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
PORTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
REUSABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEROPER-ABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 8 of 10)

FUNCTIONAL REQUIREMENTS / ENGINEERING REQUIREMENTS	B.6.d	B.7	B.7.a	B.7.b	B.7.c	B.7.d	B.7.e	B.7.f	B.7.g	B.8	G.1	G.2	G.3
USABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
MAINTAINABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
TESTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
FLEXIBILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
PORTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
REUSABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEROPER-ABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 9 of 10)

FUNCTIONAL REQUIREMENTS / ENGINEERING REQUIREMENTS	G 4	G 7	G 8	G 9	G 10	G 10 a	G 10 b	G 10 c	G 10 d	G 10 e	G 10 f	G 10 g	G 11
USABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
MAINTAINABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
TESTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
FLEXIBILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
PORTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
REUSABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEROPER-ABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-2. Mapping of Applicable Requirements to the FD
(Part 10 of 10)

ENGINEERING REQUIREMENTS \ FUNCTIONAL REQUIREMENTS	G 12	G 13	G 13 a	G 13 b	G 13 c	G 13 d	G 14	G 14 a	G 14 b	G 14 c	G 14 d	G 14 e	G 15	G 16
USABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●	●
MAINTAINABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●	●
TESTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●	●
FLEXIBILITY	●	●	●	●	●	●	●	●	●	●	●	●	●	●
PORTABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●	●
REUSABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●	●
INTEROPER-ABILITY	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-3. Allocation of Applicable FD Requirements to the SRS
(Part 1 of 6)

FUNCTIONAL REQUIREMENTS \ ENGINEERING REQUIREMENTS	INT-1.001	INT-1.002	INT-1.003	INT-1.004	INT-1.005	INT-1.006	INT-1.007	CAP-1.1	CAP-1.2	CAP-1.3	CAP-1.4	RIN-1.001	RIN-1.002	RIN-1.003	RIN-1.004
A.1						●	●	●				●	●	●	
A.2						●	●	●	●	●		●	●	●	●
A.3						●		●				●	●	●	
A.4						●		●				●	●	●	
A.5						●		●				●	●	●	
A.6						●	●	●		●		●	●	●	
A.7						●	●	●	●			●	●	●	●
A.8						●		●				●	●	●	
A.9						●	●	●				●	●	●	
A.10						●		●				●	●	●	
A.11	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.1	●														
B.2	●														
B.2.a	●														
B.2.b	●														
B.2.c	●														
B.2.c.1	●														
B.2.c.2	●														
B.2.c.3	●														
B.2.c.4	●														
B.3	●														
B.4	●														

Table 3-3. Allocation of Applicable FD Requirements to the SRS
(Part 2 of 6)

FUNCTIONAL REQUIREMENTS \ ENGINEERING REQUIREMENTS	INT-1.001	INT-1.002	INT-1.003	INT-1.004	INT-1.005	INT-1.006	INT-1.007	CAP-1.1	CAP-1.2	CAP-1.3	CAP-1.4	RIN-1.001	RIN-1.002	RIN-1.003	RIN-1.004
B.6	●														
B.6.a	●														
B.6.b	●														
B.6.c	●														
B.6.d	●														
B.7	●														
B.7.a	●														
B.7.b	●														
B.7.c	●														
B.7.d	●														
B.7.e	●														
B.7.f	●														
B.7.g	●														
B.8	●														
G.1					●										
G.2					●										
G.3					●										
G.4					●										
G.7					●										
G.8					●										
G.9					●										
G.10					●										

Table 3-3. Allocation of Applicable FD Requirements to the SRS
(Part 3 of 6)

FUNCTIONAL REQUIREMENTS \ ENGINEERING REQUIREMENTS	INT-1.001	INT-1.002	INT-1.003	INT-1.004	INT-1.005	INT-1.006	INT-1.007	CAP-1.1	CAP-1.2	CAP-1.3	CAP-1.4	RIN-1.001	RIN-1.002	RIN-1.003	RIN-1.004
G.10.a					●										
G.10.b					●										
G.10.c					●										
G.10.d					●										
G.10.e					●										
G.10.f					●										
G.10.g					●										
G.11					●										
G.12					●										
G.13					●										
G.13.a					●										
G.13.b					●										
G.13.c					●										
G.13.d					●										
G.14					●										
G.14.a					●										
G.14.b					●										
G.14.c					●										
G.14.d					●										
G.14.e					●										
G.15					●										
G.16					●										

Table 3-3. Allocation of Applicable FD Requirements to the SRS
(Part 4 of 6)

FUNCTIONAL REQUIREMENTS \ ENGINEERING REQUIREMENTS	RIN-1 005	SIZING	TIMING	DESIGN CONSTRAINTS	CORRECTNESS	RELIABILITY	EFFICIENCY	INTEGRITY	USABILITY	MAINTAINABILITY	TESTABILITY	FLEXIBILITY	PORTABILITY	REUSABILITY	INTEROPER-ABILITY
A.1		●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.3		●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.4		●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.5	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.6		●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.7		●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.8		●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.9		●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.10		●	●	●	●	●	●	●	●	●	●	●	●	●	●
A.11	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.1		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.2		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.2.a		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.2.b		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.2.c		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.2.c.1		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.2.c.2		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.2.c.3		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.2.c.4		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.3		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.4		●	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-3. Allocation of Applicable FD Requirements to the SRS
(Part 5 of 6)

FUNCTIONAL REQUIREMENTS \ ENGINEERING REQUIREMENTS	RIN-1 ODS	SIZING	TIMING	DESIGN CONSTRAINTS	CORRECTNESS	RELIABILITY	EFFICIENCY	INTEGRITY	USABILITY	MAINTAINABILITY	TESTABILITY	FLEXIBILITY	PORTABILITY	REUSABILITY	INTEROPERABILITY
B.6		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.6.a		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.6.b		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.6.c		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.6.d		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.7		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.7.a		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.7.b		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.7.c		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.7.d		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.7.e		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.7.f		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.7.g		●	●	●	●	●	●	●	●	●	●	●	●	●	●
B.8		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.1		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.2		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.3		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.4		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.7		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.8		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.9		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.10		●	●	●	●	●	●	●	●	●	●	●	●	●	●

Table 3-3. Allocation of Applicable FD Requirements to the SRS
(Part 6 of 6)

FUNCTIONAL REQUIREMENTS \ ENGINEERING REQUIREMENTS	RIN-1 005	SIZING	TIMING	DESIGN CONSTRAINTS	CORRECTNESS	RELIABILITY	EFFICIENCY	INTEGRITY	USABILITY	MAINTAINABILITY	TESTABILITY	FLEXIBILITY	PORTABILITY	REUSABILITY	INTEROPER-ABILITY
G.10.a		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.10.b		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.10.c		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.10.d		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.10.e		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.10.f		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.10.g		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.11		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.12		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.13		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.13.a		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.13.b		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.13.c		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.13.d		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.14		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.14.a		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.14.b		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.14.c		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.14.d		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.14.e		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.15		●	●	●	●	●	●	●	●	●	●	●	●	●	●
G.16		●	●	●	●	●	●	●	●	●	●	●	●	●	●

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 4. QUALIFICATION REQUIREMENTS

This section specifies the qualification methods to be used to ensure that the CSCI requirements of sections 3 and 5 have been satisfied.

4.1 Qualification Methods

This paragraph discusses the qualification methods to be used to ensure that all requirements of the Human Interface CSCI have been satisfied. The methods utilized shall satisfy the requirements described in the Software Quality Program Plan (SQPP) and in section 5 (Formal Qualification Testing) of the Software Development Plan (SDP). The specific methods to be utilized are as follows and a qualification cross-reference table appears as table 4-1:

a. Demonstration - the use of stubs and drivers to permit the functional operation of specific program unit(s) to ensure that the function to be performed is done so correctly.

b. Test - the execution of specific program unit(s) utilizing test data to ensure that the algorithmic logic performs correctly, in accordance with established test procedures.

c. Analysis - the verification and interpretation of the results obtained from the various methods described in this paragraph whereby the Quality Assurance (QA) Manager shall analyze the accumulated results to ensure that quality assurance standards are maintained.

d. Inspection - the visual review of source code and documentation to ensure that both coding standards and documentation guidelines are followed.

e. Reviews - the use of In-Process Reviews (IPRs), Initial Operational Capability (IOC), and Final Operational Capability (FOC) reviews to ensure that software development fulfills the defined requirements.

4.2 Special Qualification Requirements

No special qualification requirements are applicable for this CSCI.

Table 4-1. Qualification Cross-Reference Table

REQUIREMENT NAME	CM IDENTIFIER	SECTION 3 PARAGRAPH	QUALIFICATION METHODS*	QUALIFICATION LEVEL**
GUI Processing	CAP-1.1	3.2.1	A,D,I,R,T	1,2
Help Facility	CAP-1.2	3.2.2	A,D,I,R,T	1,2
Error Handling	CAP-1.3	3.2.3	A,D,I,R,T	1,2
Motif Tools	CAP-1.4	3.2.4	A,I,R,T	1,2

* Qualification Method

A - Analysis

D - Demonstration

I - Inspection

R - Reviews

T - Test

** Qualification Level

1 - Configuration Item

2 - System Integration

3 - System Installation

SECTION 5. PREPARATION FOR DELIVERY

The Human Interface CSCI (CSCI-1) shall consist of all completed FOCs integrated into an operational system along with any corrected deficiencies. The preparation of the CSCI for delivery shall include, but not be limited to, the following (on a Sun workstation):

a. Recompile and relink all source code and create object and executable files

b. Provide necessary documentation to support the CSCI

c. Provide magnetic media (1.44 Mb disks or 1/4" tapes) copies of both source code and executable files in support of the CSCI

d. Provide a list of all known deficiencies

e. Provide a listing of all source programs involved in the preparation of the CSCI.

Documentation to be delivered with the CSCI includes the Software Development Folders (SDFs) for the CSCI and a Version Description Document (VDD). Furthermore, that portion of a Software Release Bulletin (SRB) appropriate to the CSCI shall also be produced and delivered.

The release media is UNCLASSIFIED and shall be accompanied by a delivery letter.

THIS PAGE INTENTIONALLY LEFT BLANK

SECTION 6. NOTES

This section contains information of general interest that aids in understanding this specification. Specifically, document references to include both source and issue date are provided as well as a terms and abbreviations paragraph.

6.1 Document References

The following references were used in the preparation of this specification:

a. American National Standards Institute (ANSI), Programming Language C, ANSI X3.159-1989, New York, NY, 16 December 1989

b. Department of Defense (DOD), Defense System Software Development, Department of Defense Standard, DOD-STD-2167A, Washington, D.C., 29 February 1988

c. DOD, Software Requirements Specification, Data Item Description (DID), DI-MCCR-80025A, Washington, D.C., 29 February 1988

d. Joint Data Systems Support Center (JDSSC), Documentation Standards and Publications Style Manual, Procedures Manual (PM) 1-90, Washington, D.C., 1 August 1990

e. JDSSC, Functional Description for the Graphic Information Presentation System (GIPSY), <reference>, Washington, D.C., 1 February 1988

f. JDSSC, Software Development Plan (SDP) for the Mapping and Graphic Information Capability (MAGIC), SDP 2-90, Washington, D.C., 1 November 1990

g. JDSSC, Software Quality Program Plan for the Mapping and Graphic Information Capability (MAGIC) (Draft), <reference>, Washington, D.C., 23 July 1990

h. JDSSC, Software Standards and Procedures Manual for the JNGG Graphics Program, Technical Memorandum (TM) 405-90, Washington, D.C., 1 December 1990

i. National Technical Information Service (NTIS), The X Window System, Massachusetts Institute of Technology (MIT)/Laboratory for Computer Science (LCS)/Technical Report (TR)-368, Cambridge, MA, November 1986

j. Open Software Foundation (OSF), OSF/Motif Users Guide, Revision 1.0, Englewood Cliffs, NJ, Prentice Hall, 1990.

6.2 Terms and Abbreviations

The following terms, abbreviations, and acronyms specific to this document are listed below:

ADP	Automated Data Processing
Allocated Baseline	The initially approved documentation describing an item's functional and interface characteristics that are allocated from those of a higher level CI; specified by MIL-STD-480B
ANSI	American National Standards Institute
C	The C programming language as specified by ANSI Standard X3.159-1989
CAP	Configuration identifier prefix used to designate a capability
CI	Configuration Item
COTS	Commercial Off-The-Shelf
CSCI	Computer Software Configuration Item
Cyclomatic Complexity	A software metric that provides a quantitative measure of the logical complexity of a program.
DAFC	Directive Action File Control; the GIPSY file that holds all essential common areas and end-of-memory arrays during a GIPSY user session
DBMS	Database Management System
DI	Data Item
DID	Data Item Description
DMS	DeLorme Mapping System
DOD	Department of Defense
DOD-STD	Department of Defense Standard
DTIC	Defense Technical Information Center
FD	Functional Description as specified by DID # DI-IPSC-80689 of DOD-STD-7935A
FOC	Final Operational Capability
GIPSY	Graphic Information Presentation System
GUI	Graphical User Interface
INT	Configuration identifier prefix used to designate an external interface
IOC	Initial Operational Capability
IPR	In-Process Review
IPSC	Information Processing Standards for Computers
IRS	Interface Requirements Specification as specified by DID # DI-MCCR-80026A of DOD-STD-2167A
JDSSC	Joint Data Systems Support Center
JN	NMCS ADP Directorate
JNG	General Applications Division
JNGG	Information Systems Branch; the OPR for MAGIC development
JTSA-P	Administrative Control Branch; Pentagon Technical Resource Center, Room MF612A

LCS ---------- Laboratory for Computer Science; part of MIT
MAGIC ---------- Mapping and Graphic Information Capability
Mb ---------- Megabyte; 1,024,000 bytes of data
MCCR ---------- Mission-Critical Computer Resources
MIT ---------- Massachusetts Institute of Technology
Module ---------- In the MAGIC environment, a C language function
NMCS ---------- National Military Command System
NTIS ---------- National Technical Information Service; formerly the National Bureau of Standards
OPR ---------- Office of Primary Responsibility
OSF ---------- Open Software Foundation
PM ---------- Procedures Manual
QA ---------- Quality Assurance
RAM ---------- Random Access Memory
Rational ---------- The R1000 Ada language-based development platform manufactured and sold by Rational Corporation
RIN ---------- Configuration identifier prefix used to designate an internal interface requirement
SDF ---------- Software Development Folder
SDP ---------- Software Development Plan as specified in DID # DI-MCCR-80030A of DOD-STD-2167A
SPARC ---------- Scalable Processor Architecture
SQL ---------- Structured Query Language as defined in ANSI X3.135-1986
SQPP ---------- Software Quality Program Plan as specified in DID # DI-QCIC-80572 of DOD-STD-2168
SRB ---------- Software Release Bulletin
SRS ---------- Software Requirements Specification as specified by DID # DI-MCCR-80025A of DOD-STD-2167A
SSPM ---------- Software Standards and Procedures Manual as specified by DID # DI-MCCR-80011 of DOD-STD-2167
TM ---------- Technical Memorandum as specified by JDSSC PM 1-90
TR ---------- Technical Report
TSS ---------- Time Sharing System
VDD ---------- Version Description Document as specified by DID # DI-MCCR-80013A of DOD-STD-2167A
WWMCCS ---------- Worldwide Military Command and Control System
X Windows ---------- A device-independent and network-transparent windowing protocol for graphics workstations developed at MIT and copyrighted in 1984
Xlib ---------- The library of low-level routines comprising the X Windows System
xterm ---------- A terminal emulator provided with the X Window System which emulates either the Tektronix 4014 or the VT102 terminal type

THIS PAGE INTENTIONALLY LEFT BLANK

DISTRIBUTION

Addressees	Copies
JDSSC Codes	
JTSA-P (Record and Reference Set)	3
JNGG	30
Defense Technical Information Center (DTIC) Cameron Station, Alexandria, VA 22304-6145	2

	35

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	1 December 1990	Software Requirements Specification

4. TITLE AND SUBTITLE
Software Requirements Specification for the Mapping and Graphic Information Capability (MAGIC)
Volume I - Human Interface CSCI

5. FUNDING NUMBERS

6. AUTHOR(S)
David W. Hall and D. Rene' Cozad

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Joint Data Systems Support Center (JDSSC)
Room BF670C, The Pentagon
Washington, D.C., 20301-7010

8. PERFORMING ORGANIZATION REPORT NUMBER
SRS 1-90

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This Software Requirements Specification (SRS) specifies the engineering and qualification requirements for the Human Interface CSCI of the Mapping and Graphic Information Capability (MAGIC). Furthermore, this specification will be used as the basis for the design and formal testing of that CSCI.

The SRS is divided into 3 major sections. These sections cover Engineering Requirements (Section 3), Qualification Requirements (Section 4), and Preparation for Delivery (Section 5).

14. SUBJECT TERMS
N/A

15. NUMBER OF PAGES
56

16. PRICE CODE
N/A

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	None